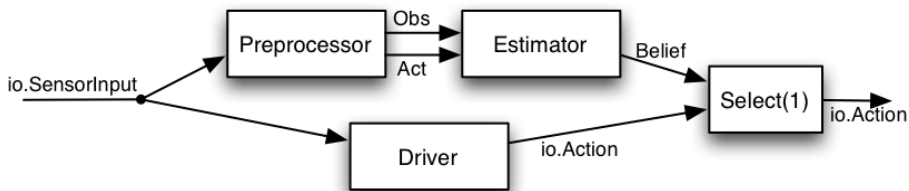


问题 Wk.12.2.3: 本地化

之前（在 Wk.11.1.7 中），您考虑过如何计算某个世界的理想声纳读数。请参考该问题，以了解有关机器人沿走廊移动并使用单侧声纳估计其在走廊中的位置的设置。现在我们将了解定位系统的整体结构。在您熟悉结构后，我们将要求您手动模拟一个简单的案例。

该系统的架构如下：



系统中的关键模块包括：

- **预处理器**：这[状态机](#) 以以下实例作为输入 [io.SensorInput](#) 并生成输出对（观察、行动），适合输入到状态估计器。
- **估算器**：这个状态机是一个[实例seGraphics.StateEstimator](#)，本质上就是您之前实现的状态估计器。
- **司机**：此状态机是[这移动.移动至固定姿势班级](#) .可用于使机器人直线前进。

以下是系统关键参数的描述（以及我们将在本导师问题中使用的值）：

- 观察次数 = 10：一个正整数，描述我们将声纳范围划分为的离散值的数量。我们假设声纳读数在 0.0 到声纳这是最大可靠声纳读数。
- 状态数 = 10：一个正整数，描述离散值的数量，我们将划分机器人的可能 x 坐标系。我们假设机器人的 x 位置在范围内 $x_{\text{最小}}$ 到最大。
- $x_{\text{最小}}=0.0$ ：尽可能小 x 机器人在世界上的坐标（以米为单位）。
- $x_{\text{Max}}=10.0$ ：尽可能大 x 机器人在世界上的坐标（以米为单位）
- $y=0.5$ ：固定是机器人在世界上的坐标（以米为单位）。
- $\text{sonarMax}=1.5$ ：最大声纳读数（以米为单位）。

我们感兴趣的是估计机器人的 x 坐标，基于其观察和动作的顺序。因此，我们将以与观察相同的方式处理状态空间，并将机器人的可能范围划分为 x 坐标为 n 数固定间隔，并让状态机器人的索引是其真实间隔的索引 x 坐标下降。设 Δx 是间隔的“宽度” x 构成状态的空间。

预处理器

预处理器是一个具有以下输入和输出的状态机：

- **输入**：实例 [io.SensorInput](#)。

- **输出：**一对（观察，行为），每个都是一个整数索引。观察是离散声纳读数的索引，行为是离散运动的指标。

为了使这个问题适合我们的状态估计框架，我们将离散化通过将可能的声纳值范围划分为观察次数固定长度间隔，并将值编码为其所属间隔的索引。

因为我们希望状态估计器与控制机器人动作的另一台机器并行运行，但不直接依赖于它，所以我们将根据观察机器人的里程表推断出“动作”。回想一下，里程表读数是相对于机器人在世界上的任意起始位置（我们不知道）。因此，里程表仅适用于估计相对位移。

让 x_t 被机器人观察到 t 时间协调 w 我们会说机器人在某个时间采取的‘行动’ $t-1$ 是 $x_t - x_{t-1}$ 。我们需要将动作空间离散化，因此我们实际上会将动作报告为一个四舍五入的间隔数：它是 $\text{int}(\text{round}((x_t - x_{t-1}) / w))$ ，在哪里 w 是状态区间的宽度（即可能 x 坐标除以离散状态数）。

因为我们的状态估计过程包含观察和随后的转变，预处理器从步骤输出观察结果至关重要 $t-1$ 以及步骤之间发生的动作 $t-1$ 和 t 。

状态机首次启动时，不会有任何先前的观察或里程表可用，因此我们将仅生成一个输出没有任何。我们将对状态估计器进行特殊修改，这样如果机器的输入是没有任何，则根本不会进行任何状态更新。

估算器

这估算器是作为状态估计器的状态机实例；其输入和输出为：

- **输入：**任何一个没有任何或一对（观察，行为），每个都是一个整数索引。观察是时间离散声纳读数的索引 t 和行为是时间开始的离散运动的索引 t 。
- **输出：**信念状态，表示为距离 DDist 超过可能的离散 x 机器人在世界上的位置。如果输入是没有任何，那么输出的信念状态与前一时间步骤的信念状态（或者如果这是第一个时间步骤，则与初始状态）相同。

为了制作估计器状态机，需要构造一个[随机SM](#)包含初始分布、局部化问题的转移和观测模型。我们可以让起始分布均匀。

状态估计器的起始状态只是 SSM 的初始信念状态。为了得到下一个值，我们执行以下步骤[贝叶斯证据](#)与观察相结合，并应用全概率定律进行观察。

```

类 StateEstimator (sm.SM) :
    def __init__(self, 模型):
        自我.模型 = 模型
        自我.startState = 模型.startDistribution

    def getNextValues (self, state, inp) :
        如果 inp == None: 返回 (state, state)

```

```

(o, i) = inp
sGo = dist.bayesEvidence(状态, self.model.observationDistribution, o) dSPime =
dist.totalProbability(sGo,
                                自我.模型.过渡分布(i))
返回 (dSPime, dSPime)

```

观察模型

请记住，观测模型是一个条件分布，即一个以状态为输入并返回可能观测值的分布的过程。状态将是我们的离散状态索引之一。

在这个问题中，假设完美的观察模型，即在每个状态下，声纳都会返回离散化该状态的理想读数（您在上一个问题中编写了代码来计算这些读数）。

过渡模型

类似地，转换模型是一个以动作作为输入并返回过程的过程；该过程以起始状态作为输入，并返回结果状态的分布。

假设完美的转换模型，即给定一个动作 Δ （一个整数）和一个状态 s （也是一个整数），结果状态是 $s + \Delta$ 但要剪裁以保持合法范围内 $0 \leq s < \text{状态数}$ 。

提醒：因为我们的状态估计过程包含观察和随后的转变，预处理器从步骤输出观察结果至关重要 $t-1$ 以及步骤之间发生的动作 $t-1$ 和 t 。

在这个问题中，我们假设每个状态的理想（离散）声纳读数是：

理想 = (5, 1, 1, 5, 1, 1, 1, 5, 1, 5)

填写输出值预处理器和估算器以下前三个时间步骤的状态机。对于非零概率，请输入小数点后3位数字（输入小数值或分数）。

如果预处理器的输出是没有任何，进入没有任何这里的两个框中都有；但是当您在设计实验室 13 中实现这个状态机时，您的机器应该输出一个 None。

1.预处理器（时间 0）：

- 输入：一个实例io.SensorInput：
 - 声纳 = (0.8, 1.0, ...) 里程计 = Pose(1.0, 0.5, 0.0)
- 输出：一个元组（观察，行为）；如果输出是没有任何，进入没有任何在两个盒子里。
 - 观察=
 - 行为=

2.预处理器（时间 1）：

- 输入：一个实例io.SensorInput：
 - 声纳 = (0.25, 1.2, ...) 里程计 = Pose(2.4, 0.5, 0.0)

○ 输出：一个元组（观察，行为）；如果输出是没有任何， 进入没有任何在两个盒子里。

■ 观察=	<input type="text"/>
■ 行为=	<input type="text"/>

3.预处理器（时间 2）：

○ 输入：一个实例io.SensorInput：

- 声纳 = (0.16, 0.2, ...) 里程计 = Pose(7.3, 0.5,
- 0.0)

○ 输出：一个元组（观察，行为）；如果输出是没有任何， 进入没有任何在两个盒子里。

■ 观察=	<input type="text"/>
■ 行为=	<input type="text"/>

4.估计量（时间 0 时）：

○ 输入：（观察，行为）预处理器输出的输出元组：机器人状态的概率分布

○ （x 索引）

<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		

5.估计量（时间 1）：

○ 输入：（观察，行为）预处理器输出的输出元组：机器人状态的概率分布

○ （x 索引）

<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		

6.估算器（时间 2 时）：

○ 输入：（观察，行为）预处理器输出的输出元组：机器人状态的概率分布

○ （x 索引）

<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		